UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/021,260	12/19/2001	Carsten Driesner	30014200.1018	7197

58328 7590 12/07/2009

SUN MICROSYSTEMS
C/O SONNENSCHEIN NATH & ROSENTHAL LLP
P.O. BOX 061080
WACKER DRIVE STATION, WILLIS TOWER
CHICAGO, IL 60606-1080

EXAMINER	
CAO, DIEM K	
ART UNIT	**PAPER NUMBER**
2194	

MAIL DATE	DELIVERY MODE
12/07/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/021,260	DRIESNER ET AL.	
	Examiner	Art Unit	
	DIEM K. CAO	2194	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *16 July 2009*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-35* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-35* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-35 are pending. Applicant has amended claims 1, 12, 22, 23, 34 and 35.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1 – 14 and 20 – 35 are rejected under 35 U.S.C. 103(a) as being unpatentable over Coulouris (Coulouris, George, Jean Dollimore and Tim Kindberg, "Distributed Systems Concepts and Design," Second Edition, Addison-Wesley, 1994.) in view of Fidge (Fidge, Colin, "Logical Time in Distributed Computing Systems," Computer, Volume 24, Issue 8, August 1991, pages 28 – 33, ISSN: 0018-9162; retrieved from IEEE.) and Nazarathy et al. (US 6490727 B1; "Nazarathy").

As to claim 1, Coulouris teaches a method in a data processing system for synchronizing calls at a client in a server and client system, comprising the steps of:

receiving from the server a plurality of service calls generated by a plurality of threads executed at the server [page 326 ¶ 3 and page 135 ¶ 6 – 7; see also page 150 ¶ 1 – page 151 ¶ 5 and page 12 ¶ 1];

receiving a synchronization call from the server, said synchronization call being a separate and different type of call from the service calls and indicating that one of said plurality of threads executed at the server has changed and indicating a number of service calls generated by said plurality of threads at the server prior to the thread change [page 326 ¶ 3, the count of events indicates the number of calls; p. 396 ¶5 – 6; p. 397 ¶5]; and

placing at least one of said service calls associated with said synchronization call into a wait position, the at least one of the service calls corresponding to the changed thread, when said number of service calls indicated in said synchronization call and said number of service calls executed at the client prior to receiving said synchronization call differ [page 342 ¶ 6, the timestamps can be based on logical clocks, which counts events, such as requests, to maintain ordering of events and requests, and a multicast message … has to be placed on the hold-back queue until it can be delivered in causal order].

As to claim 1, Fidge also teaches a method in a data processing system for synchronizing calls at a client in a server and client system, comprising the steps of:

receiving from the server a plurality of service calls generated by a plurality of threads executed at the server [Fig. 1; page 29 ¶ 5 and page 30 Rule H, the processes perform events, including communication actions];

receiving a synchronization call from the server, said synchronization call being a separate call from the service calls and indicating that one of said plurality of threads executed at the server has changed and indicating a number of service calls generated by said plurality of threads at the server prior to the thread change [page 30 Rules B and F,

the "ticks" count events, such as messages. When the threads block in rule F, the threads have changed and exchange the logical time, which indicates the number of calls]; and

placing at least one of said service calls associated with said synchronization call into a wait position, the at least one of the service calls corresponding to the changed thread, when said number of service calls indicated in said synchronization call and said number of service calls executed at the client prior to receiving said synchronization call differ [page 30 col. 3 ¶ 7 – 9; page 33 ¶ 3 – 4; requests are processed in the same order that they were made, not received].

Although Coulouris and Fidge fail to specifically state that synchronization is done with a thread changes, Coulouris discloses thread switching [page 173 ¶ 5] and the synchronization counts events and sends the count to the proper locations [page 326 ¶ 3]. Also, Coulouris teaches buffering and sending when a thread blocks (changes) [page 151 ¶ 5]. It would have been obvious to one of ordinary skill in the art at the time Applicant's invention was made to use vector timestamps because regardless of whether the processes run on separate machines or on the same machine, this method provides a method of synchronizing calls in the system. Furthermore, Fidge teaches synchronization [page 30 Rule F]. It would have been obvious to one of ordinary skill in the art at the time Applicant's invention was made to combine these references because both address the same problem of ordering with similar solutions, counters, timestamps and vector timestamps.

Coulouris further teaches that the calls can include different types of calls, such as read calls and write calls (p. 396 ¶5 – 6; p. 397 ¶5). Therefore, a later synchronizing call can be

separate and a different type than the preceding service calls. However, Nazarathy more clearly teaches that it is known to use separate messages with timestamps to provide synchronization functionality. It would have been obvious to one of ordinary skill in the art at the time Applicant's invention was made to combine these teachings because Coulouris teaches providing synchronization functionality with timestamps and Nazarathy teaches additional features and methods of synchronization with the use of timestamps that can be applied to the teaches of Coulouris to produce expected results.

As to claim 2, Coulouris teaches said service calls are associated with said synchronization call by one of including respective identifiers into said at least one of said synchronization call and said service calls [page 135 ¶ 5 – 6], and indicating one of a specific reception sequence and order of service of said service calls and said at least one synchronization call at the client [page 326 ¶ 3]. Fidge also teaches the use of identifiers [page 30 Rule A].

As to claim 3, Coulouris teaches said receiving steps include receiving a first call sequence of a plurality of call sequences from the server, said first call sequence including a first synchronization call and at least one service call from a first thread, said first synchronization call including a first server call counter value indicating a first number of service calls executed at the server prior to the first synchronization call [page 342 ¶ 5; page 151 ¶ 5];

said method further comprising the step of:

comparing said first server call counter value with a client call counter value, said client call counter value indicating a second number of service calls executed at the client

prior to receiving said first synchronization call [page 342 ¶ 6 including listed criteria; page 152 ¶ 2 – 3 and page 298 ¶ 6]; and one of:

executing said first number of service calls of said first call sequence and counting said executed first number of service calls using a client call counter value, if said client call counter value and said first server call counter value coincide [page 342 listed steps and ¶ 6]; and

placing said first call sequence into a wait position, if said client call counter value and said first server current call counter value differ [page 342 ¶ 6].

As to claim 4, Both Coulouris and Fidge also disclose that said service calls are generated asynchronously [Coulouris: page 152 ¶ 3; Fidge: page 30 Rules F – H].

As to claim 5, Coulouris teaches the additional steps of:

determining whether a second call sequence in a wait position is available, said second call sequence including a plurality of service calls from a second thread executed at the server and a second synchronization call including a second server call counter value indicating a third number of service calls executed at the server prior to said second synchronization call [page 342 ¶ 5 through criteria list of ¶ 6];

wherein if said second call sequence in a wait position is not available, waiting to receive further service calls and synchronization calls [page 342 ¶ 6]; and

wherein if said second call sequence is available, determining that said second server call counter value coincides with said client call counter value, and executing said

third number of service calls of said second call sequence and incrementing said client counter value for each executed third number of service calls [page 342 ¶ 5 through criteria list of ¶ 6].

As to claim 6, Coulouris teaches waiting for a third call sequence to be received from the server unit, the third call sequence including a third synchronization call including a third server call counter value coinciding with said client call counter value [page 342 ¶ 5 through criteria list of ¶ 6].

As to claim 7, Coulouris teaches said call sequences are received as groups included into packets from the server, each group being generated upon one of a timer signal at the server, a synchronous call at the server, and a synchronization call at the server [page 151 ¶ 5].

As to claim 8, Coulouris teaches said synchronization call and said service calls are received in an arbitrary order [page 325 ¶ 1].

As to claim 9, Coulouris teaches said service calls from said plurality of threads at the server are executed in corresponding threads at the client [page 135 ¶ 7].

As to claim 10, Coulouris teaches said first server call counter value indicates a total number of service calls at the server executed prior to a current service call and requires communication with the client [page 326 ¶ 3]; and

wherein said client call counter value indicates a total number of service calls executed at the client and involves communication with the server [page 326 step 3].

As to claim 11, Coulouris teaches each of said service calls form the server includes at least one of:

obtaining instructions to display information on a display of the client [page 149 ¶ 10];

rendering instructions;

storing instructions to store information at the client; and

information on processing results from the server.

As to claim 12, the limitations are rejected for the same reasons as limitations in claim 1. For example, the limitation of claim 12 that recites transmitting service calls by a server is rejected for the same reasons and references as the limitation in claim 1 that recites receiving service calls from the server.

As to claims 13, 14 and 20, see the rejection of claims 2, 4 and 11.

As to claim 21, Coulouris teaches a synchronization call is further generated upon an occurrence of one of the group consisting of: a timer signal [page 151 ¶ 5]; a predetermined number of service calls; and a synchronous call.

As to claims 22 and 23, see the rejections of claims 1 and 12.

As to claims 24 – 33, see the rejections of claims 2 – 11.

As to claim 34, Coulouris teaches a data processing system for synchronizing calls in a client and server system, the data processing system comprising:

a client computer comprising a memory including a client program and a first processor that runs said client program [Figure 1.1; page 326 ¶ 2 – 3];

a server computer comprising a memory including a server program and a second processor that runs said server program [Figure 1.1; page 326 ¶ 2 – 3]; and

a network connecting said client computer and said server computer [Figure 1.1].

See the rejections of claims 1 and 12 regarding the functions of the client and server programs.

As to claim 35, see the rejections of claims 1 and 12.

3. Claims 15 – 19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Coulouris in view of Fidge and Nazarathy as applied to claim 12 above, and further in view of Liedtke (Liedtke, Jochen, "Improving IPC by Kernel Design," ACM Symposium on Operating Systems Principles, Proceedings of the fourteenth ACM Symposium on Operating Systems Principles, ACM Press, 1994; pages 175 – 188.).

As to claim 15, Coulouris teaches the steps of:

generating a current service call by a first thread executed at the server [page 326 ¶ 3];

wherein, if said first thread and said second thread differ, generating a first synchronization call including a server call counter value indicating a number of service calls executed at the server prior to said current service call and transmitting said first synchronization call to the client [page 326 ¶ 3 and step 2], for enabling the client to synchronize an execution of a plurality of service calls from at least said first thread and said second thread [page 326 – 327, vector clock update algorithm]; and

counting said current service call using said server call counter value if said first thread identifier and said second thread identifier do not differ [page 326 ¶ 3].

Coulouris fails to specifically teach determining and comparing thread identifiers and carrying out the appropriate action depending on whether or not the identifiers differ. Liedtke teaches using unique identifiers to distinguish between threads [page 180 section 5.3.1]. It would have been obvious to one of ordinary skill in the art at the time Applicant's invention was made to use the thread IDs to determine if two calls were made by the same thread or different threads in order to confirm a thread change because the thread IDs are unique, which provides a way to distinguish between threads (see motivation below). Comparing thread IDs provides confirmation that a thread change occurred regardless of whether or not a change was requested or expected. The appropriate response can then be performed. See also the explanation in Coulouris regarding vector timestamp information for different processes and buffering calls [page 326 ¶ 3; page 151 ¶ 5], providing motivation to determine the source of calls.

It would have been obvious to one of ordinary skill in the art at the time Applicant's invention was made to combine these references because Coulouris teaches the use of multiple processes and Liedtke teaches identifiers that can be used to distinguish between processes.

As to claim 16, see the rejection of claim 7.

As to claim 17, Coulouris teaches said synchronization call includes said second thread identifier of said second thread, and said number of service calls include a thread identifier of each thread generating said service call [page 326 ¶ 3, the vector timestamp provides information regarding the other processes. See also the rejection of claim 15 regarding thread identifiers.] and wherein said synchronization call and said number of service calls are transmitted to the client in an arbitrary order [page 325 ¶ 1].

As to claim 18, see the rejection of claim 9.

As to claim 19, Coulouris teaches said server call counter value indicates a total number of service calls requiring communication with the client executed at the server, prior to the current service call [page 326 ¶ 3]. The timestamp vector provides counts from all processes.

Response to Arguments

4. Applicant's arguments filed 7/16/2009 have been fully considered but they are not persuasive.

In response to Applicant's arguments that the combination of Coulouris and Fidge fails to teaches the limitations of claims 1, 12, 22, 23, 34 and 35, examiner respectfully disagreed because, first, claims 1, 12, 22, 23, 34 and 35 are rejected under the combination of Coulouris, Fidge, and Nazarathy, second, Applicant fails to provide any reasons why the cited passages do not teach the claim's limitations.

Therefore, the arguments are not persuasive.

Conclusion

5. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

6. Any inquiry concerning this communication or earlier communications from the examiner should be directed to DIEM K. CAO whose telephone number is (571)272-3760. The examiner can normally be reached on Monday - Friday, 7:30AM - 4:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Hyung Sough can be reached on (571) 272-6799. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/DIEM K CAO/
Primary Examiner
Art Unit 2194

DC
December 4, 2009